ANALYSIS OF CHILI FRUITS DETECTION FROM STEREO CAMERA IMAGES USING YOLOV5

MUHAMMAD SHAHMIN NASHRI BIN SHAHRUL AZLAN

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ANALYSIS OF CHILI FRUITS DETECTION FROM STEREO CAMERA IMAGES USING YOLOV5

MUHAMMAD SHAHMIN NASHRI BIN SHAHRUL AZLAN

This report is submitted in partial fulfilment of the requirements for the degree of Bachelor of Electronic Engineering with Honours

Faculty of Electronic and Computer Engineering
Universiti Teknikal Malaysia Melaka

2022

DECLARATION

I declare that this report entitled " ANALYSIS OF CHILI FRUITS DETECTION FROM STEREO CAMERA IMAGES USING YOLOV5" is the result of my own work except for quotes as cited in the references.

Signature : .....

Author : MUHAMMAD SHAHMIN NASHRI BIN SHAHRUL AZLAN

Date : June 2022

APPROVAL

I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of Bachelor of Electronic Engineering with Honours.

Signature : .....

TS. DR. MUHAMMAD NOORAZLAN SHAH BIN ZAINUDIN
Deputy Dean (Student Development)
Faculty of Electronics and Computer Engineering
Universiti Teknikal Malaysia Melaka
Hang Tuah Jaya
76100 Durian Tunggal, Melaka, Malaysia

Supervisor Name : .....

Date : 20 June 2022

DEDICATION

To my adored parents, lectures, family, and colleagues.

ABSTRACT

Chili is one of the fruits that has become as essential for cooking mostly for Malaysian's people. Eating chili provides an additional spicy taste in ancient times. There is evidence of archaeological discovery sites located in south-western Ecuador where they add a chili as an additional ingredient of food since 600 years ago, and it was one of the most important plant for growing areas on the American continent at the time as a chili farm. Less accurate for chili maturity and labour intensive, an automated approach for chili picking is prevalent. In many image recognition problems, 2D images is used. However, due to the lack of image information such depth, 2D images is considered impractical to be applied in real environment. Hence, this work aims to detect and recognize the chili fruits in order to estimate their maturity level and also for pursuing picking process. This work is expected to identify the shape and maturity of a chilli through the color using YoLov5. This work also is a part of our intention to develop a semi-autonomous chili picking robot.

ABSTRAK

Cili adalah salah satu buah yang menjadi keperluan untuk memasak kebanyakannya untuk rakyat Malaysia. Makan cili memberikan rasa pedas tambahan pada zaman dahulu. Terdapat bukti tapak penemuan arkeologi yang terletak di barat daya Ecuador di mana mereka menambah cili sebagai bahan tambahan makanan sejak 600 tahun lalu, dan ia merupakan salah satu tumbuhan terpenting untuk kawasan penanaman di benua Amerika pada masa itu sebagai ladang cili. Kurang tepat untuk kematangan cili dan intensif buruh, pendekatan automatik untuk memetik cili adalah lazim. Dalam banyak masalah pengecaman imej, imej 2D digunakan. Walau bagaimanapun, disebabkan kekurangan maklumat imej kedalaman sedemikian, imej 2D dianggap tidak praktikal untuk digunakan dalam persekitaran sebenar. Oleh itu, kerja ini bertujuan untuk mengesan dan mengenali buah cili bagi menganggar tahap kematangannya dan juga untuk mengikuti proses memetik. Karya ini diharapkan dapat mengenal pasti bentuk dan kematangan cili melalui warna menggunakan YoLov5. Kerja ini juga merupakan sebahagian daripada hasrat kami untuk membangunkan robot pemetik cili separa autonomi.

ACKNOWLEDGEMENTS

First and foremost, Alhamdulillah and thanks to Allah, the Almighty, for giving me to complete this Projek Sarjana Muda (PSM) successfully. Without the guidance of our Almighty, this project would not be completed on time. This thankful is also gratitude to FKEKK and internal PJP Grant from Universiti Teknikal Malaysia Melaka (UTeM) for the financial support. In this short period, there are a lot of obstacles and challenges that we must overcome. Throughout this project, i would like to express our utmost gratitude toward my project supervisor, Dr. Muhammad Noorazlan Shah Zainudin and Dr. Wira hidayat Bin Mohd Saad, for providing me the title and all the other information regarding the title proposed for my Projek Sarjana Muda (PSM) and all their germinal ideas and guidance throughout this project completion. Without my supervisor, i would not fully grasp the concept of this Projek Sarjana Muda (PSM), which could not have hindered our progress. Besides, i would like to thank our family and colleagues that always give a helping hand whenever necessary either directly or indirectly. Finally, I would like express my gratitude to everyone who has helped and supported me throughout Projek Sarjana Muda (PSM).

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

CNN	:	Convolution Neural Network
RGB	:	Red, green and blue
YOLO	:	You Only Look Once
mAP	:	mean Average Precision
RNN	:	Recurrent Neural Networks
ANN	:	Artificial Neural Network
AI	:	Artificial Intelligence

CHAPTER 1

INTRODUCTION

1.1 Project Background

Depth camera gives the object information such as shape, localization, classification, and distance in real world by identifying the intensity of image to shows the distance of the object captured from a viewpoint[1]. The color information in depth image gives the information about the distance of object from viewpoint. In most digital cameras, an output images are a produced in 2D grid of pixels where the information, x and y axes. Initially, every pixel of images has a value associated with it that being called RGB which is red, green, and blue. The value of attribute produces from each pixel is from 0 to 255 to represent the color code, for example, the black has the point of (0,0,0) and pure bright red would be (255,0,0)[2]. A depth camera on the other hand, has an additional pixel value which have a different numerical value associated with them. An additional information presents the distance of the object

from the camera or also called as depth information. Some depth cameras have both RGB and a depth system (D), which provide a pixel with all four values, or RGBD.

There are a several methods for calculating a depth, depending on the chosen an optimal operating condition. The conditions for calculation the depth is depending on user preferences such: How far that user need to see? What sort of accuracy that user need? Can it operate with multiple object? Can it operate for the outdoors? For instances, stereo depth camera uses infrared light onto a scene to improve the accuracy of the images[3]. Stereo depth camera has two sensor which is left imager and right imager that spaced a small distance apart, then from these two sensors, the depth camera will compare the distance from both. The two sensors used are depth and RGB sensors. These sensors work by improving correspondence between the two different data streams and to match the field of view between the depth sensors and the RGB sensor. Since the distance between two sensors is known, the depth information is obtained[4].

The integration of image recognition and object detection practices are frequently used in various industries such as agriculture, medical, security, etc. Image recognition identifies the objects or scenes contained within an image, while object detection identifies the instances and locations of those objects[5]. Image recognition can be used to automate such time-consuming tasks and the time taken to process the images more quickly and accurately than manual approach[6]. Image recognition is a critical technique used in a wide variety of applications and serves as the primary motivation for an invention of artificial intelligence such as deep learning for categorizing images according to their characteristics. This is particularly advantageous in e-commerce applications such as image retrieval and recommender systems. In the field of computer vision, object detection has undergone a rapid revolution. Due to its

involvement in both object classification and object localization, it is one of the most difficult topics in the field of computer vision. In simple terms, the objective of this detection technique is to determine the location of objects within a given image, referred to as object localization, and the category to which each object belongs, referred to as object classification[7].

Most agriculture industries start to use an automatic technique for fruits to implement the recognition using deep learning. The model will train the network in a supervised manner, with images of the fruits serving as the input and labels for the fruits serving as the output. Following successful training, the Convolutional Neural Network (CNN) is one of the prominent models which able to predict the fruit's according to its label accurately. This idea is also can be used to develop a model which capable of recognizing and predicting the name of a fruit. Sometime when need to recognize thousands of fruit images in a short period of time, there are a variety of applications could be applied for fruit recognition. For chili as an example, deep learning CNN is used to recognize its types and categories[8].

The lifetime of chili fruits does not last long if the process of picking is not done properly. This chili would quickly being rotten if the picking process is too late. The maturity level of these chili can be known in 6 categories based on its color. The first categories are immature, the immature chili is in light green in color, and it takes a week for it to change the color into dark green and shiny. The second is mature in dark green color and shiny, they are also more durable than red chilies. The third category is quite mature, this category will change the color from green into red (start to change), it can be last stored for a week at room temperature. The fourth category is also quite mature, where chili in red color and exceeds the green color (changed in 50%). The green color in blackish and this chili cannot be stored for long period of

time. The fifth category is still the same, which is quite mature, at this stage the red color has changed completely. The color is shiny and bright red and can last 1 to 2 days. The last category is over mature and usually will be used as a seed[9].

Stereo camera works in a same way to how human use two eyes by looking for depth perception. Our brains will calculate the difference between each eye to get the depth information. Objects that closer to eyes will appear to move significantly from eye to eye (or sensor to sensor), where an object in the far distance would appear to move very little. Stereo cameras can be used to create stereo views and three-dimensional images, as well as for range imaging. The distance between the lenses in a typical stereo camera (known as the intra-axial distance) is approximately 6.35 cm, though a longer base line (greater inter-camera distance) produces more extreme 3-dimensionality[10]. 3D images that adhere to the stereo camera theory can also be created more affordably by taking two images with the same camera but moving the camera a few inches left or right. If the image is edited in such a way that each eye sees a different image, the image appears to be three-dimensional. Although this method has issues with objects moving between views, it works well with still life.

Deep learning has established itself as a highly effective tool due to its capacity to handle large amounts of data. Hidden layer techniques have surpassed traditional techniques in popularity, particularly in pattern recognition. CNN are one of the most widely used model from deep neural networks categories[11]. As an example, CNN would recognize handwritten digits, detection of type of cancers, recognizing the face, etc. For handwritten recognition it primarily used in the postal sector to read zip codes, pin codes, and other unique identifiers. The critical point to remember about any deep learning model is that it requires a large amount of data and a significant amount of computing resources to train. CNN are a subclass of deep neural networks that are

most frequently used to analyze visual imagery in deep learning. Deep learning has proven in variety of applications, including image and video recognition, image classification, image segmentation, medical image analysis, and natural language processing. CNN are specialized multilayer perceptron. Multilayer perceptron is referred to fully connected networks, in which each neuron in one layer is connected to every neuron in the following layer[12]. Due to their "complete connectivity," these networks are prone to overfitting data. The input to a CNN is a tensor of the form (number of heights X input inputs X input channels X input width). One of the many fascinating uses of convolutional neural networks is image classification. Aside from simple picture categorization, computer vision presents other exciting difficulties, with object detection being among the most intriguing. YOLO ("You Only Look Once") is an efficient method for real-time object detection. Unlike previous object detection methods, which repurposed classifiers to do detection, YOLO proposes the usage of an end-to-end neural network that simultaneously predicts bounding boxes and class probabilities. YOLO produces state-of-the-art results in object detection by using a fundamentally different approach than existing real-time object detection algorithms.

1.2 Problem Statement

Classification and localization

- In most object detection problems, the process for determining the object's position generally referred to as the object localization task, is hard when 2D images are used. Not only for classifying those objects, but the detection of the correct position is crucial for implementing the detection process in real-environment situation

Object detection

- Object detection is not only able to accurately classify from its position and localize an object from its background, it's also needs to be incredibly fast at prediction time to meet the demands of video processing.

Multiple spatial scales and aspect ratios

- In any applications of object detection, object that is going to be detected may appear in a wide range of sizes and aspect ratios which is contributed to the difficulties of the process.

1.3 Objectives

1. To label the chili fruits images captured by using RGB stereo images
2. To train and analyze the object detection of labelled images using YoLov5 in term of detection accuracy.
3. To evaluate the performance of detection for different chili colors in terms of mean Average Precision (mAP).

1.4 Scope of work

This work only tackles the process of detection and identifying an object as chili fruits without estimating its maturity sizes. Matlab is used to calibrate the image captured from stereo camera. Intel Real sense SDK2.0 needs to be installed with Matlab developer package to get the Matlab wrapper. Matlab wrapper brings Intel Real Sense viewer function into Matlab. Matlab is used to capture the RGB and depth image. Using the makesense.ai in web browser to label the red and green chili. After that, training and validate the labeling image on google Colaboratoy by using

YOLOv5 function. Lastly, testing the data on the demo video to see whether red and green chili can be detected or not including accuracy for every chili.

1.5 Thesis Outline

This report is divided into five chapters. The first section is an introduction, in which the project summary, problem statement, objectives, and scope of work are explained. Chapter 2 includes information about the project that can be found in reference books, on the internet, in journals, or from other sources of information. Chapter 3 will discuss the robot motion limitation on the maximum and minimum angle of rotation for each degree of freedom. Chapter 4 will discuss the results and discussion in greater detail, while Chapter 5 will conclude this project and make some recommendations for future work.

CHAPTER 2

BACKGROUND STUDY

2.1 Introduction

In this chapter, a review is conducted to explore and gather more resourceful information which is relevant to this work. Various research papers, journals, online resources such as E-books is used to carried out a comprehensive study on multiple theories and background studies. Generally, the study focuses on the vision for developing chili picking robot. Therefore, some background study related to the development of the robot to detect and picking up the chili from the chili plant is reviewed.

2.2 Deep learning

The most successful results in the area of image recognition and classification, most of the works have been reported using artificial neural networks. These networks serve as the foundation of deep learning models. Deep learning is a subset of machine learning algorithms that employ multiple layers of processing units and also is able to solve the non-linear problems. Each level layer slightly abstracts and composites the input data. Deep neural networks have surpassed other machine learning algorithms in terms accuracy of performance in many patterns' recognition and classification domain. In the field of image recognition, deep neural networks, CNN, have been demonstrated to produce an outstanding performance.

2.3 YOLO

You Only Look Once is what the acronym YOLO stands for. "You Only Look Once" This is a piece of software that scans a picture and identifies the various things that are in it (in real-time). The process of object detection in YOLO is approached as a regression problem, and the results provide information about the class probabilities of the images that were detected. Convolutional neural networks (CNN) are utilized by the YOLO algorithm in order to accomplish real-time object detection. In order to identify objects, the algorithm needs to perform only one forward propagation through a neural network, as the name of the algorithm suggests [14].

2.3.1 YOLOv1

YOLOv1 is a model for detecting objects that only has one stage. The process of object detection is conceptualized as a regression problem with spatially distinct bounding boxes and associated class probabilities. One single evaluation allows for the prediction of bounding boxes and class probabilities directly from full images

using a single neural network[15]. Due to the fact that the entire detection pipeline is a single network, it is possible to directly optimize for improving detection performance end-to-end. When attempting to predict each bounding box, the network draws on features from the entire image. Additionally, it predicts all of the bounding boxes for an image simultaneously across all of its classes. This indicates that the network makes decisions about the entire image and each of its components at the global level. YOLO is an innovative method for the detection of objects. Classifiers have been repurposed in previous research on object detection to perform detection. This does work as described above, but it has many limitations, and as a result, the use of the YOLO v1 is restricted as a result of these limitations. It was unable to locate small objects if they were grouped together in any way. If the image has dimensions that are different from the trained image, this architecture had trouble generalizing the objects in the image. The primary challenge lies in the identification of specific locations within the input image. Instead, consider object detection as a regression problem that pertains to spatially distinct bounding boxes and the probabilities associated with each class. One single evaluation allows for the prediction of bounding boxes and class probabilities directly from full images using a single neural network. Due to the fact that the entire detection pipeline is a single network, it is possible to directly optimize it for improved detection performance end-to-end. This architecture can achieve very high speeds. The YOLO model at its core is capable of processing images in real time at a rate of 45 frames per second. Fast YOLO, a more compact version of the network, manages to process an astounding 155 frames per second while still achieving twice the mean Average Precision (mAP) of other real-time detectors. When compared to other detection systems that are considered to be state-of-the-art, YOLO has a higher rate of localization errors but has a lower likelihood of predicting

false positives in the background. YOLO will eventually learn very general representations of the objects it encounters. When it comes to generalizing from natural images to other domains like artwork, it outperforms other detection methods such as DPM and R-CNN [16].

2.3.2 YOLOv2

YOLOv2, also known as YOLO9000, is a model for detecting real-time objects in a single stage. YOLO9000's meaning is that the system is able to identify more than 9000 different types of objects [17]. It is superior to YOLOv1 in a number of respects, including the utilization of Darknet-19 as a backbone, batch normalization and utilization of a high-resolution classifier, fine-grained features, multi-scale training, as well as utilization of anchor boxes to predict bounding boxes, among other things. The most significant changes that have been made to this version make it better, faster, and more advanced so that it can meet the requirements of the Faster R-CNN. This is another object detection algorithm that employs a Region Proposal Network in order to recognize the objects that are present in the image input. The input layer is normalized through a process known as batch normalization, which involves slightly modifying and scaling the activations. The stability of the neural network is improved as a result of the use of batch normalization, which brings about a reduction in the shift in unit value that occurs in the hidden layer. The mAP of the architecture has been improved by two percent as a result of the addition of batch normalization to convolutional layers. It also assisted in regularizing the model, which contributed to an overall reduction in overfitting. The size of the inputs in YOLO v2 has been increased to 448*448 from the previous 224*224. The mAP has improved by as much as 4 percent as a result of the increase in the size of the image that is being input. This increase in input size was implemented during the training of the YOLO v2

architecture on the ImageNet dataset carried out by DarkNet 19. One of the most noticeable changes that can be seen in YOLO v2 is the introduction of the anchor boxes. This change is one of the most notable changes that can be seen. Within a single framework, YOLO v2 is able to perform classification and prediction. The bounding box can be predicted based on the information contained in these anchor boxes. One of the most important challenges that the YOLO v1 needs to overcome is the detection of smaller objects on the image, and fine-grained features are a part of this challenge. This issue has been fixed in the YOLO v2 software, which partitions the picture into 13 by 13 grid cells that are more compact than those in the earlier version of the programme. This makes it possible for the Yolo v2 to recognize or localize smaller objects in the image, in addition to being effective with respect to larger objects. Multi-scale training. n If YOLO is trained with small images of a particular object, it has issues detecting the same object on the image of a bigger size. This is because YOLO's weakness in detecting objects with different input sizes is a weakness in detecting objects with different input sizes. This issue has been addressed to a large extent in YOLO v2, which trains itself on random images of varying dimensions ranging from 320 pixels by 320 pixels up to 608 pixels by 608 pixels. This enables the network to learn and make accurate predictions about the objects based on the many different input dimensions. In conclusion, YOLO v2 makes use of the Darknet 19 architecture, which consists of 19 convolutional layers, 5 max-pooling layers, and a softmax layer for the classification of various objects. The underlying structure of Darknet 19 is outlined in the following diagram. The neural network framework known as Darknet was developed using the C programming language and CUDA. It can detect objects quickly, which is a quality that is essential for making accurate predictions in real time. Because of these advancements in a variety of areas, YOLO v2 is now superior, both

in terms of its speed and its strength. Because of Multi-Scale Training, the network is now able to recognize and categorize objects that have a variety of configurations as well as dimensions. In comparison to its predecessor, YOLO v2 is significantly superior in terms of its ability to detect smaller objects with a greater degree of precision. This capability was severely lacking in the earlier version [18].

2.3.3 YOLOv3

You Only Look Once, Version 3 (YOLOv3) is a real-time object detection algorithm that can identifies an objects in videos, live feeds, or images. In order to identify an object, YOLO analyses its features through the lens of a deep convolutional neural network. In comparison to YOLO and YOLOv2, YOLOv3 is a significant advancement. Deep learning libraries such as Keras and OpenCV are utilized in the development of YOLO[19]. Programs that use artificial intelligence rely on object classification systems in order to determine which specific objects within a class should be treated as subjects of interest. The systems separate the objects in the images into categories, where objects that share similar qualities are grouped together and other objects are ignored unless they are specifically programmed to behave differently. Real-time object detection is the primary function of the CNN known as YOLO. CNNs are a type of classifier-based system that can process incoming images as structured arrays of data and identify patterns between the images (view image below). YOLO is an advantage over other networks due to the fact that it is significantly faster while still preserving its accuracy. It gives the model the ability to examine the entire image during testing, which ensures that its predictions are based on the overall context present in the image. The "score" given to regions by YOLO and other convolutional neural network algorithms is determined by how closely those regions resemble previously defined categories. Positive detections of whatever class

a region most closely aligns itself with are attributed to high-scoring regions. First, the YOLOv3 algorithm creates a grid out of the image. Each grid cell makes a prediction regarding the number of boundary boxes (also called anchor boxes) that should be placed around objects that have a high score. Each boundary box has a respective confidence score that indicates how accurate it assumes that prediction should be, and it can only detect a single object within each bounding box. The boundary boxes are generated by clustering the dimensions of the ground truth boxes from the original dataset to find the most common shapes and sizes. R-CNN, which stands for Region-based Convolutional Neural Networks and was developed in 2015, Fast R-CNN, which is an improvement on R-CNN and was developed in 2017, and Mask R-CNN are some other comparable algorithms that are able to accomplish the same goal. YOLO, on the other hand, is trained to perform classification in addition to bounding box regression at the same time, in contrast to systems such as R-CNN and Fast R-CNN. When it comes to accuracy, speed, and architecture, OLOv2 and YOLOv3 are on completely different planets. YOLOv2 was using Darknet-19 as its backbone feature extractor, whereas YOLOv3 is now using Darknet-53. This change was made to improve the speed of the system. In terms of mAP and intersection over union (IOU) values, YOLOv3 is quick and accurate as well. It operates at a speed that is orders of magnitude faster than other detection methods while maintaining comparable performance. YOLOv3 has the ability to detect objects that are small, medium, and large in terms of their average precision (AP). It denotes that the higher the AP, the more accurate the variable [20]. Table 2.1 shows the different between YOLO v1 and YOLO v3.

Table 2.1 shows the comparison of YOLOv1, YOLOv2 and YOLOv3 for different categories of data.

Table 2.1: Difference for YOLOv1 to YOLOv3

YOLOv1	YOLOv2	YOLOv3
Used darknet framework that trained on ImageNet 1000 dataset in 30 FPS	Second version YOLO or named as YOLO9000(9000 dataset) in 40FPS	The previous version that has been improved for an incremental improvement that called YOLOv3 in 45 FPS
Cannot find small object if they are appeared as a cluster	Higher resolution classifier, Fine-grained features, multi scale training	Feature pyramid network (FPN)
Architecture found difficulty in generalisation of object if the image is of other dimensions different from the trained images	Darknet-19: Yolov2 uses darknet-19 architecture	The predecessor YOLOv2 used darknet-19 as feature extractor and YOLOv3 uses the darknet-53 network for feature extractor which has 53 convolutional layers

2.3.4 YOLOv4

When it comes to detection performance as well as superior speed, YOLOv4 significantly outperforms the other methods that are currently available. YOLOv4 describes it as an object detector that "quickly operates" and that can be trained easily

so that it can be implemented in production systems[21]. This problem is solved by YOLO version 4 is a single GPU that can trained on it to create object detector using a more manageable mini-batch size. Because of this, it is now possible to train a super-fast object detector with high accuracy using only a single GPU with the model number 1080 Ti or 2080 Ti. On the MS COCO dataset, YOLO v4 achieves state-of-the-art results at a real-time speed, with 43.5 percent AP running at 65 FPS on a Tesla V100. This is a significant improvement over previous version. The architecture of YOLOv4 consists of the CSPDarknet53 as a backbone, the spatial pyramid pooling additional module, the PANet path-aggregation neck, and the YOLOv3 head. CSPDarknet53 is a backbone that has the potential to improve CNN's capacity for learning. The spatial pyramid pooling block is added on top of the CSPDarknet53 block in order to expand the receptive field and differentiate the most important features of the context [22]. Planet is used as the method for parameter aggregation for various detector levels rather than Feature pyramid networks (FPN), which were previously utilized in YOLOv3 for the purpose of object detection. Compared to EfficientDet, which is another competitive recognition model, YOLOv4's rate of improvement is twice as fast while maintaining comparable levels of performance. In addition, when compared to YOLOv3, AP and (Frames per Second) FPS both saw increases of 10 and 12 percent, respectively. The futuristic recognizer known as YOLO boasts faster FPS and greater accuracy than other detectors currently on the market. Because the detector can be trained and used on a regular GPU, this paves the way for its widespread implementation. The accuracy of the classifier as well as the detector has been significantly improved by new features introduced in YOLOv4, which may also be utilized in other research endeavors [23].

2.3.5 YOLOv5

The acronym YOLO relates to the phrase You Only Look Once. Version 5, launched by Ultralytics in June 2020, is the most cutting-edge item detection algorithm currently available[24]. CNN is a sophisticated convolutional neural network that can recognise objects with a high degree of accuracy in real time. This approach applies a single neural network to the entire picture for processing. After the picture is divided into its component pieces, bounding boxes and probabilities are projected for each of these parts. Utilizing the predicted probability, a weight is assigned to each of these bounding boxes. The approach "just looks at the picture once" in the sense that predictions are made after only one forward propagation pass through the neural network. In other words, the method "just looks once" at the image. After non-max suppression, it then delivers the items that were detected (which ensures that the object detection algorithm only identifies each object once). YOLOv5, which is concerned with memory, has both positives and negatives associated with it. 88 percentage points more compact than the YOLOv4 and 180 percentage points quicker than the YOLOv4. In terms of frame rate, YOLOv5 runs at 140 FPS while YOLOv4 only manages 50 FPS [25]. When it comes to precision, YOLOv4 and YOLOv5 are almost identical, and the difference between the two is only 0.003 points. Since there are no official papers for the YOLOv5, we can assume that it is still in the process of being developed and that ultralytics will provide regular updates on its progress. YOLOv5 is available in 4 different sizes (s, m, l, and xl). Intuitively, the larger the network, the more parameters that can be adjusted, and the better the performance. To reiterate, having a larger number of parameters results in longer training and inference times. If one were to construct a system capable of real-time detection, they would opt for either the small or the medium-sized model. Because increasing the size of the model requires more

CUDA memory, the batch size must be decreased in order for the batch to be able to fit into memory. It is not advisable to work with batches of a small size because these batches tend to have poor batch normalization statistics. Because of this, smaller and medium-sized models are more likely to be motivated than larger models [26]. Figure 2.1 shows the comparison from various versions of YOLO v5.

Model	AP^{val}	AP^{test}	AP_{50}	$Speed_{GPU}$	FPS_{GPU}		params	FLOPS
YOLOv5s	36.6	36.6	55.8	**2.1ms**	**476**		7.5M	13.2B
YOLOv5m	43.4	43.4	62.4	3.0ms	333		21.8M	39.4B
YOLOv5l	46.6	46.7	65.4	3.9ms	256		47.8M	88.1B
YOLOv5x	**48.4**	**48.4**	**66.9**	6.1ms	164		89.0M	166.4B

Figure 2.1: YOLOv5 model[27]

2.4 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNN) are a subset of models in deep learning. Convolutional layers, pooling layers, Rectified Linear Unit (ReLU) layers, and fully connected layers is integrated to construct a complete network. In CNN architecture, each convolutional layer is structured by a ReLU layer, followed by a pooling layer, followed by one or more convolutional layers, and finally by one or more fully connected layers as shown in Figure 1.1. Notably, a CNN converts the input to a one-dimensional array, which reduces the trained classifier's sensitivity to positional changes.[28]

2.5 Comparison neural network

Table 2.2 shows the comparison of several neural network architectures in deep learning for different categories of data. This comparison involves Artificial Neural

Network (ANN), Recurrent Neural Networks (RNN) and CNN in terms of data used, recurrent connection, parameter sharing, spatial relationship and vanishing and exploding gradient[29].

Table 2.2: Comparison neural network architecture

	Artificial Neural Network (ANN)	Recurrent Neural Networks (RNN)	Convolution Neural Network (CNN)
Data	Tabular data, Image data and Text data	Sequence data, Text data and audio data	Image data
Recurrent connection	No	Yes	No
Parameter sharing	No	Yes	Yes
Spatial relationship	No	No	Yes
Vanishing and Exploding Gradient	Yes	Yes	Yes

2.6 Intel Real Sense Camera

Intel Real Sense can be as depth and tracking technologies that enables machines and devices to perceive depth. Intel's technologies are used in autonomous drones, robots, and smart home devices, among other mass market products. Real Sense's product portfolio consists of Vision Processors, Depth and Tracking Modules, and Depth Cameras, all of which are supported by an open source, cross-platform SDK that simplifies camera support for third-party software developers, system integrators, ODMs, and OEMs. The Intel RealSense Vision Processor D4 series is a family of

vision processors built on 28 nanometer (nm) process technology for real-time stereo depth data computation. They employ a depth algorithm that enables more precise and extended depth perception than has previously been possible. Snapshot is a camera that is intended to be integrated into tablets and possibly smartphones. It is intended for use in taking photographs and subsequently refocusing, measuring distances, and applying motion photo filters. The refocus feature differs from that of a plenoptic camera in that RealSense Snapshot captures images with a large depth of field, initially focusing on the entire scene, and then selectively blurring parts of the image based on their distance.[30]

2.7 Image detection and classification

Image or object detection is a computer technology that processes the image and detects objects. There are the few differences in image detection and image classification. If want to classify image as items, then it is classification but if want to locate the item like to find out the number of objects from the image, then it is image detection. Image detection works like human's brain. For example, in one image, there are consisting of several items on the table, and need to detect the book on the table, our brain will decide whether there is there any book identified on the table. By using an Artificial Intelligence (AI) to perform the image detection and classification, it will detect any object by labeled data image, item coordinates, class labels and location. When the image is selected with different location object, item also change their coordinates and the sized. It will help AI to understand even though this object can be located in different places on the image with different sizes, it would not change its class. For image classification, it is a process of labeling object in the image. Neural network has to process and detect different image with different objects and classify the type of the item on the image. There are different types of AI solutions for image

classification and recognition. CNN for example applies the filters to detect certain features in the image. The way of CNN works by fully relies on the type of the filter applied. So, when applying AI to image classification, it will provide the network with as many different features as possible for analyzing their values upon training.

2.8 Computer Vision

Computer vision is defined as methods that enable computers to examine and extract the contents of images or multidimensional data in general to aid in the solution of a particular vision problem, such as pattern classification. Computer vision applications include those that automate processes such as those performed by a robot or a vehicle. Computer vision systems are also used in agriculture, such as a system for monitoring plants that functions similarly as human eye which can identify, measure, and track the target for image processing. With the advancement of computer vision, this technology has been widely applied to agricultural automation and has played a critical role in its development. When the analysis involving a huge number of variables typically requires a large amount of memory and computation power, or the use of a classification algorithm might over-fits the training samples and performs unwell when inferring the new samples. Oftentimes, features in pattern recognition contain information about the grey scale, texture, shape, or context. In image processing or computer vision, an initial pattern measurement or a subsequence of a pattern measurement is transformed into a new pattern feature. The process of classifying the object's higher-level information is called pattern classification. Thus, extracted features are used to classify or categorize the object. Additionally, it is used to automatically identify objects in an image by developing a classification algorithm[31].

2.9 Depth Camera

To gather fruits, the robot arm does not only move in one dimension but in three-dimension space. A generic RGB camera might determine the location of an object on a plane. However, the distance between the camera and the object is difficult to determine with a single camera. If a single camera moves linearly down a slide rail, the calibration error of the motor and the slide rail equipment must be considered. This study developed harvesting equipment for installation on a transport truck, comprising a depth camera and a central control host[32]. The research included many technologies, including picture capture and object identification. After collecting a high-quality image, the object detection model would be used to determine the maturity of the subject. The depth camera was then used to compute the three-dimensional space's coordinates. The RealSense cameras were designed to create RGB and depth pictures with aligned pixels. In this study, concentrate on fruit recognition at the level of each individual image and so the photos at different camera heights and over neighbouring places are not handled for this project.

2.10 Training data set

Sample tagging of the object to be detected in the image would be done manually. Then the picture and the tagged data would be fed to the neural network for training. However, the input data significantly affects machine learning. It would be easy for an unskilled image to identify the item incorrectly or not at all. In order to evaluate the training efficacy of a model, the obtained data will be separated into training datasets and validation datasets.

2.11 Feature and representation for classification

Features are distinguishing physical properties of an object that allow it to be distinguished from others. The fruit has various physical features, particularly colour,

texture, shape, and size, that may be employed as a feature for successful categorization. The fruit exhibits a great deal of intraclass and interclass diversity. Changes in colour, texture, and form characterise differences across classes, but intraclass variations are typically far more subtle and difficult to discern. Computer-based representation aspects are another level of this difficulty. Numerous research have been conducted on feature representation. Investigations have demonstrated that a single characteristic cannot be relied upon to effectively classify fruits and vegetables or things in general[33]

2.12 Related Work

This table 2.3 discusses some literature regarding the localization and detection some object using stereo camera in real time by applying neural network function. Some of the finding in locate and detect the object by size and some of them using colour to differentiate the object without interference.

Table 2.3: Literature Review

Year	Author	Literature review	Finding
2018	Scott Helmer and David Lowe	Using Stereo for Object Recognition	fusing 2D image information and depth information from stereo images into one model for localization, particularly in the case of contour-based objects
2018	Abhipray Paturkar	Discussion about Apple Detection for Harvesting Robot	1. To detect the fruits by differentiating them from the background 2. To locate them accurately

		Using Computer Vision	3. To detect the apples under change in illumination scenario
2019	Christian Hofmann, Florian Particke, Markus Hiller and Jorn Thielecke	Discussion about Object detection, Classification and Localization by Stereo Cameras	• An algorithm for detecting objects based on depth maps • Simultaneous use and fusion of three different object detection algorithms • Object localization based on stereo cameras
2014	Deepika Jaswal, Sowmya.V, K.P.Soman	Discuss about image classification and application of CNN	application of Convolutional Neural Network or CNN for image classification
2012	M. Hanafi Ani, Amelia Azman	Disscussion about Object Distance and Size Measurement Using Stereo Vision System	measuring the distance of an object as well as the size of the object using stereo vision sensor. The method also employs a much faster algorithm so that the measurement can be done in real-time
2014	Edy winarno agus, Harjoko aniati murni,	Discussion about recognition system	A face recognition system developed with real-time

	Arymurthy edi winarko	and face using stereo vision camera	distance estimation using stereo vision camera

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter describes the methodology used in this work. A detailed explanation is provided regarding the process that includes the collection data and the analysis processes used to further understand the outcome of the research. The chapter generally begins with the research design where it will show the data flow which begins from the problem refinement right up until the conclusion. A process flowchart is generally generated for conducting the research from start till the end. Besides, the area of study and the method used for the conducting the project has also been discussed. A good research design enables the author to address the right method and present meaningful findings.

3.2 Flowchart

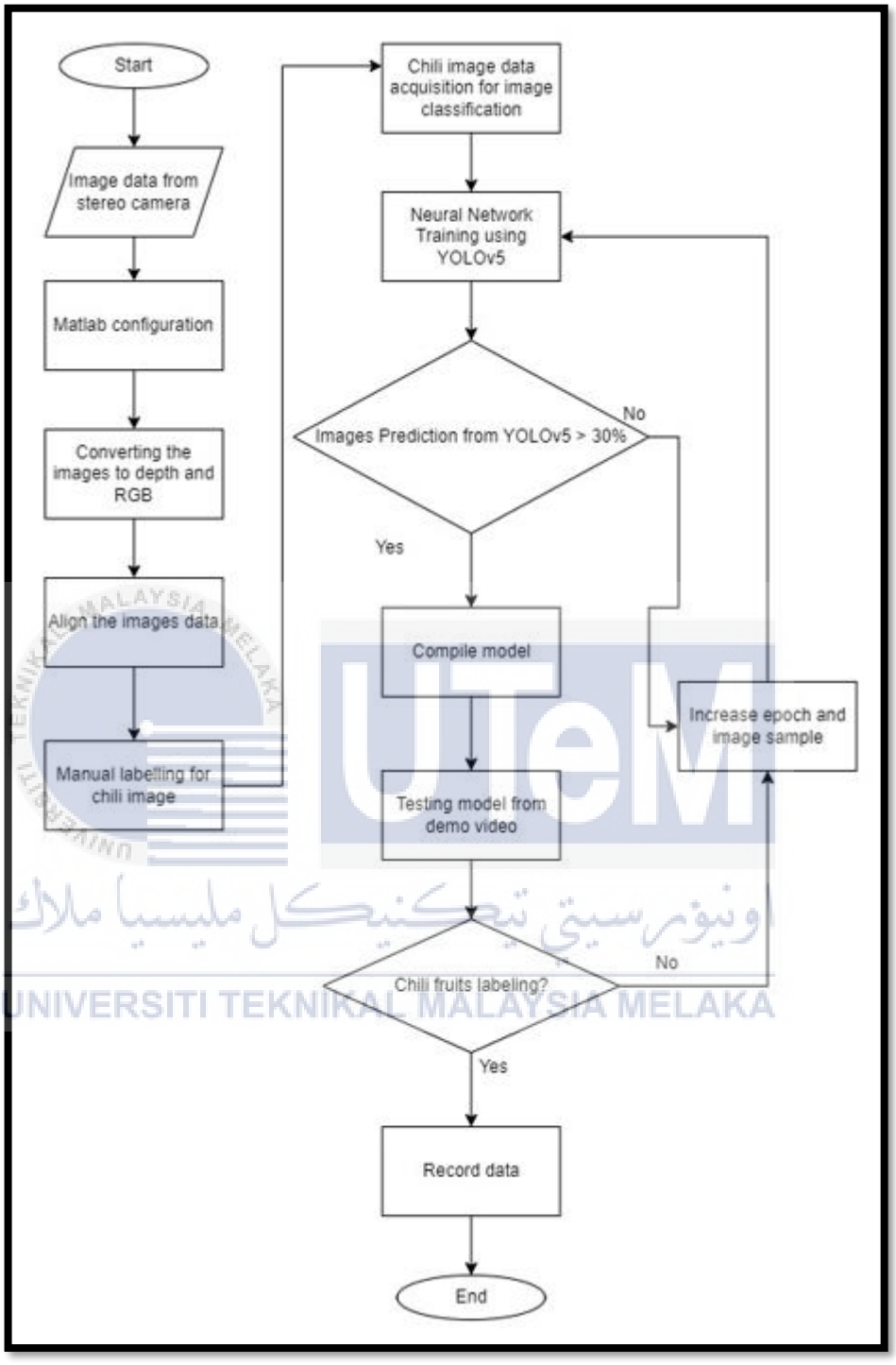

Figure 3.1: Flowchart for this project

Figure 3.1shows the detailed research methodology presents all the processes involved in this work. This refers the process from the beginning of the project until the end.

The project begins by snapping and collecting 3D stereo image of chili fruit from the chili plant with the stereo camera. Then, using the matlab to get the RGB and depth images. Matlab also being used to align the images data to get the same pixel for both images. After that, the process labeling by using makesense.ai in web browser to label the green and red chili. Next, using the YOLOv5 method neural network is used to train and test the image of chili to determine whether the chili could be detected accurately in terms of detection accuracy.

3.3 Detail Description Flowchart

- **Image data from stereo camera**

  Stereo camera is used to capture live view data and collect stereo image chili fruits from the plant.

- **Matlab configuration**

  Matlab configuration for Matlab wrapper that function to bring Intel Real Sense SDK 2.0 into Matlab for enabling the configuration of Real Sense function.

- **Converting the image to depth and RGB**

  The live view image from stereo camera is shown and it will convert into depth and RGB image by using the Matlab wrapper. The depth image is uses to measure the distance chili plant while the RGB images allows the color to vary within a single character

- **Align the images data**

Both inputs for RGB and depth images are aligned to get the same pixels. Alignment the images is a technique of warping both images for ensuring the features of two image lines up perfectly

- **Manual labelling for chili image**

Using the makesense.ai in browser enables to label the chili in multiple videos or image sequences. This tool use to Simultaneously label multiple time-overlapped signals representing the same scene. Write, import, and use custom automation algorithms to automatically label data image and evaluate the performance of your label automation algorithms by using a visual summary.

- **Chili image data acquisition for image classification**

Image acquisition is processes that can be broadly defined as the action of retrieving an image from some camera. The image captured usually completely unprocessed. For implementation, 60% of this image is used and randomly choose for training while 40% is used for validation.

- **Neural network training using YOLOv5**

Training 60% from the dataset by using YOLOv5. If the mAP is greater than 80% then it will go to testing model, if not then need to add more images and increase the epoch to get the higher accuracy.

- **Testing the data**

Testing the data by using demo video to see if the chili can be detected or not. If the chilli can be detected, then it will show a label box on the chilli either red chilli or green chilli.

3.4 Hardware of components

The hardware required in this work is explained to define the functionality and its compatibility for conduction this work.

3.4.1 Stereo Camera

Intel Real Sense camera us USB-powered camera that include wider field of view with deptsig sensor and RGB sensor. This camera can extend the distance the depth sensor up to 95mm that improves depth error less than 2% for better accuracy. To improve the RGB image and the correspondence between the depth and RGB images, the RGB sensor includes a global shutter and is matched to the depth with having an 86° field of view. The stereo camera Intel Real Sense D455 uses open-source Intel Real Sense SDK making it easy to switch from previous Real Sense cameras to D455.

3.5 Project Software

Software is a collection of instructions, data, or programs that enable computers to operate and perform specific tasks. It is the polar opposite of hardware, which refers to a computer's physical components.

3.5.1 Matlab

Matlab is a programming platform is designed specifically for student to create and analyze the system or the products. User is enabled to analyze the data, develop the

algorithm, and create models and application. In Intel Real Sense SDK2.0 provides an option in Matlab Developer Package. This function brings Intel Real Sense function to Matlab for viewer apps. The functions align is also can be used to align the images for depth map and RGB images.

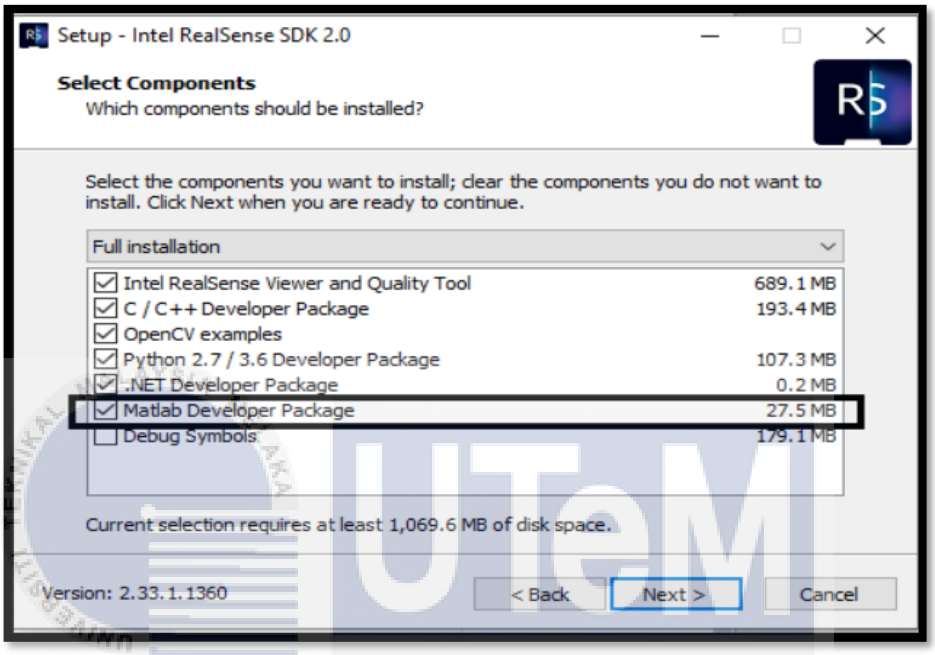

Figure 3.2: Intel RealSense SDK2.0 to Matlab

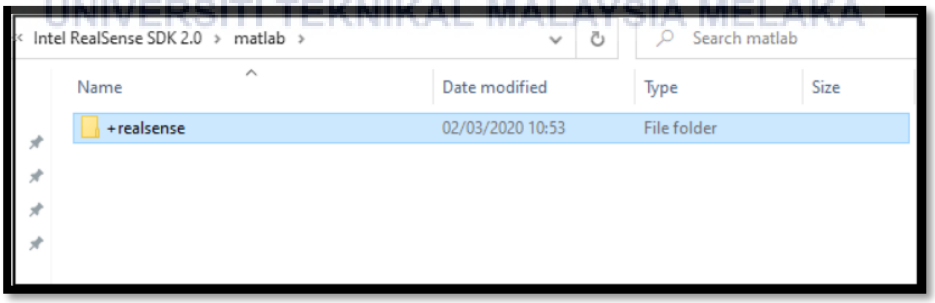

Figure 3.3: Matlab developer package

3.5.2 Intel Real Sense Viewer

Intel Real Sense Technology is a family of depth and tracking technologies that enables machines and devices to perceive depth. Intel-owned technologies are used in autonomous drones, robots, augmented and virtual reality, and smart home devices, among other broad market products. Real Sense's product portfolio consists of Vision Processors, Depth and Tracking Modules, and Depth Cameras, all of which are supported by an open source, cross-platform SDK that simplifies camera support for third-party software developers, system integrators, ODMs, and OEMs. Multiple depth and tracking technologies are supported by Intel RealSense Group, including Coded Light Depth, Stereo Depth, and Positional Tracking. The Intel RealSense Vision Processor D4 series is a family of vision processors built on 28 nanometer (nm) process technology for real-time stereo depth data computation. They make use of a depth algorithm that enables more precise and extended depth perception than was previously possible.

3.5.3 Dataset

The dataset used in this project is ‘chili localization’ obtained from the Intel realsense camera d455. This camera will connect to the Matlab apps to snap the chili image in RGB and depth image for dataset. This dataset contains more than 100 chili images. The image is an image of 2 types of chilies which is green and red colour that being recorded to do the labelling process.

3.5.4 Labelling

In labelling process, the images in RGB will be label by using the tool that can be found in web browser which is makesense.ai. All the chili in every image will label in

square box follow by the color chili. Red chili will label in red box and green chili will label in green box. This process is to increase the accuracy for the localization chili.

3.5.5 Training

A developer is required to acquire a big labelled data set and design a network architecture capable of learning the model's characteristics. This strategy is particularly effective for new applications and those with a large number of output types. Training is the set of data that utilised to train and make the model to learn any pattern in the data. In epoch, training data will be sent repeatedly to the neural network design, and the model may learn the pattern or characteristics of the data. For this project, 60% from the data set will used for the training

3.5.6 Validation

The validation set is a different data set from the training set that is utilised to validate the performance of our model during training. This procedure of validation provides information that aids in fine-tuning the model's setups. It will reveal whether or not the training is progressing in the appropriate way. After each epoch, the model is trained on the training set while concurrently being evaluated on the validation set. The primary goal of dividing the dataset into a training set and a validation set is to avoid our model from being overfit. The model excels in classifying the samples in the training set but cannot generalise or make reliable predictions outside of the training set. For this project, 40% will be used for the validation

3.5.7 Epoch

An epoch is a phrase used in neural networks to describe the number of passes the machine learning algorithm has made through the full training dataset. Typically, datasets are organised into batches, especially when the quantity of data is substantial.

The higher number of epochs the higher accuracy that will get. In this project the epoch that being used is 200 because to get more accuracy for the chili to be detected at the chili tree.

3.6 Google Colaboratory

This project uses Google Collaboratory software to perform the coding process. This software was chosen because it is open source and can be used by anyone online. Since this project uses the Pyhton programming language, Google Colab has been used because it is a web-based Python editor that allows anyone to write and run arbitrary Python code. In addition, Google Colab can also save space on the use of computer graphics processing unit (GPU) to run the coding of this project because it provides free access to these resources. Nvidia K80s, T4s, and P100s are frequently accessible GPU in Colab.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter concentrates in two images which is RGB and Depth image. Depth images only be used to determine and capture the chili plant. RGB images are used to label the chili to get the chili localization. There are two types of chili that need to label which is green and red chili. After that, the label chili images later are used for training, and validation by using yolov5. Lastly, testing is applied to evaluate the model whether it can be localized the chili on its types.

4.2 RGB and Depth images

Intel realsense SDK2.0 is the application that needed to get the RGB and depth images. This application is used to connect the intel realsense camera with matlab

application to get this both images. RGB and depth image are obtained from the function in matlab.

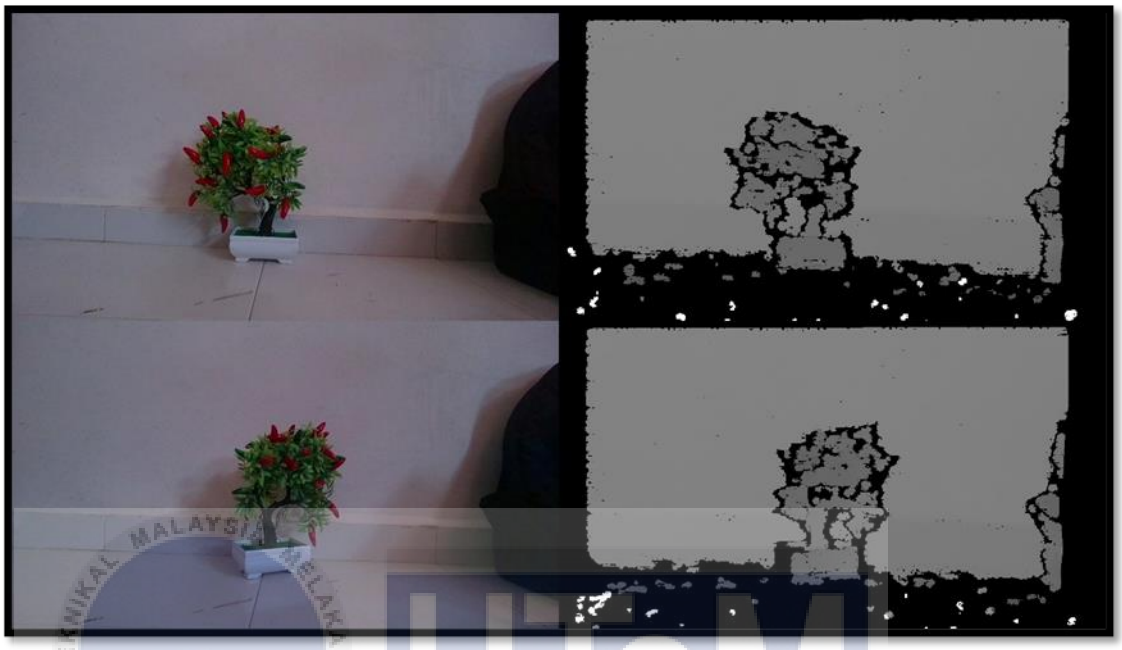

Figure 4.1: RGB (left side) and depth (right side) images chili tree

4.3 Labelling the images

RGB images that obtained from realsense intel cameras is labelled according to the type of chili. There are two ways to label the chili which is by labelling the individual chili and labelling the chillies from the chilli plant. These chillies are labelled using a rectangular box and is saved in a zip package in yolo format.

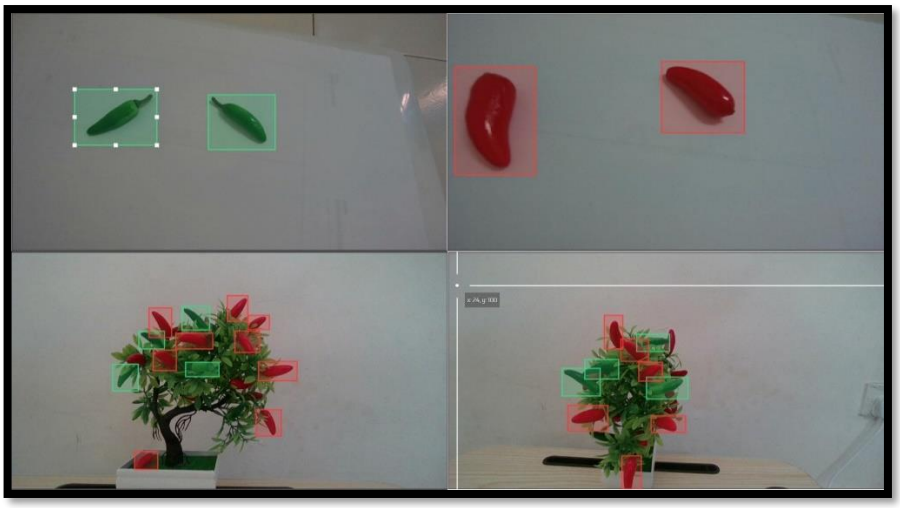

Figure 4.2: labelling the red and green chili

For both labeled chilies, a white background is used to facilitate the training process. A rectangular box is used to do this labeling process, to label this individual chili needs to follow the size of the chili and to label the chili from the plant. It's require the right size shape of the chili so that it does not conflict with the background or other chilies. After the labeling process, need to be saved the file in zip package in yolo format.

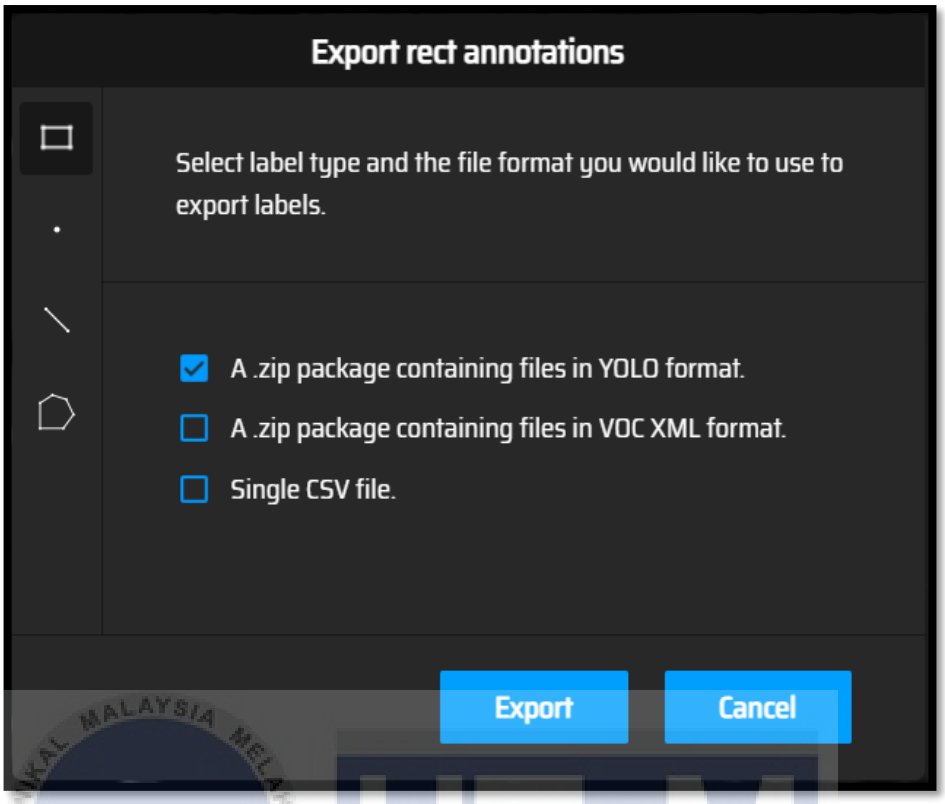

Figure 4.3: save the labeling images in yolo format

4.4 Result analysis on labelling chili individually

Green chilies is labeled as ‘green’ while red chilies is labeled as ‘red’. By using the same number of pictures for these two types of chilies and also using batch 16, epoch 50 and also the weight is yolov5s for training the images, the results can be shown in Figure 4.4 and 4.5.

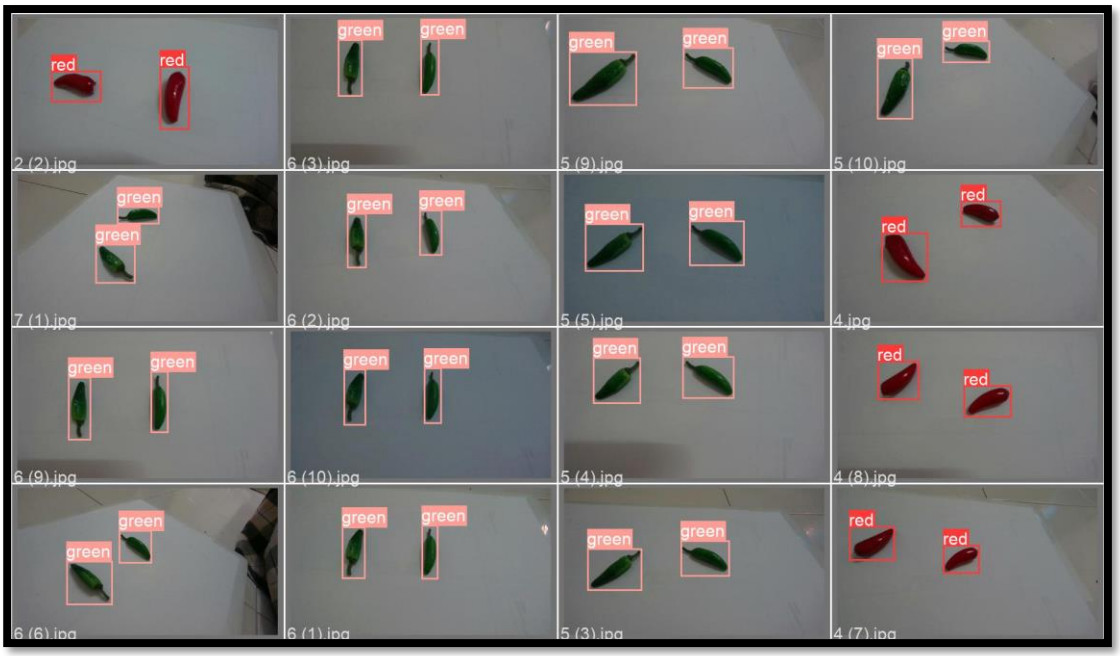

Figure 4.4: Red and green chili labeled

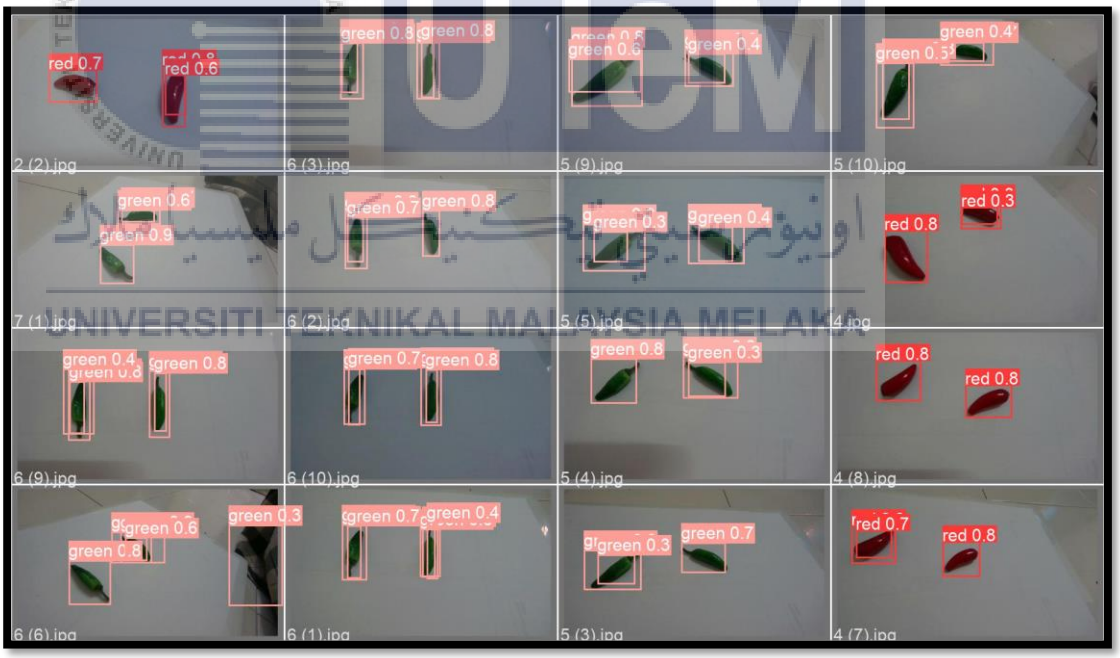

Figure 4.5: Prediction for red and green chili

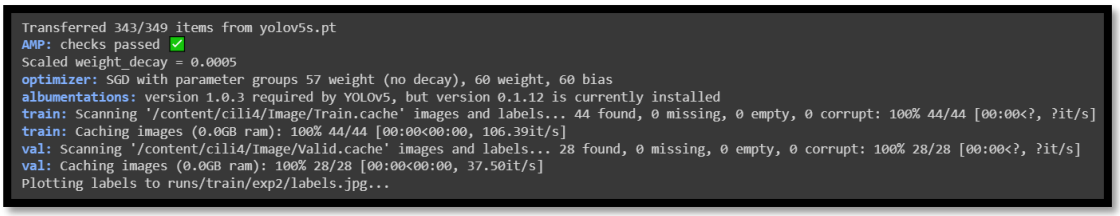

Figure 4.6: Train and validation images

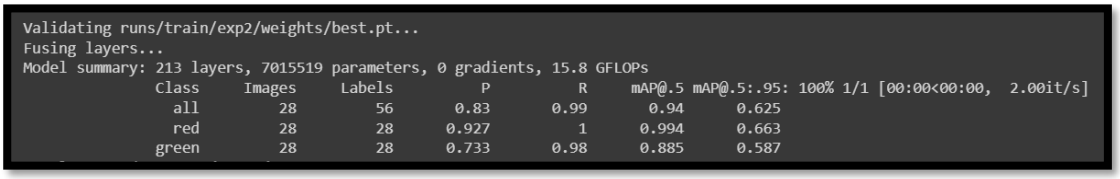

Figure 4.7: Accuracy for both red and green chili

As seen in the result prediction, for the accuracy of red chilies it is higher compared to green chilies. Due to its color representation, the color of red chilies are very bright while green chilies seem to be black in terms of color. However, these two chilies can be detected and there is no conflict between the two chilies. An average accuracy for red chilies is 0.994/94% while for green chilies it is 0.885/85%. All images that have been labeled are used for training and validation.

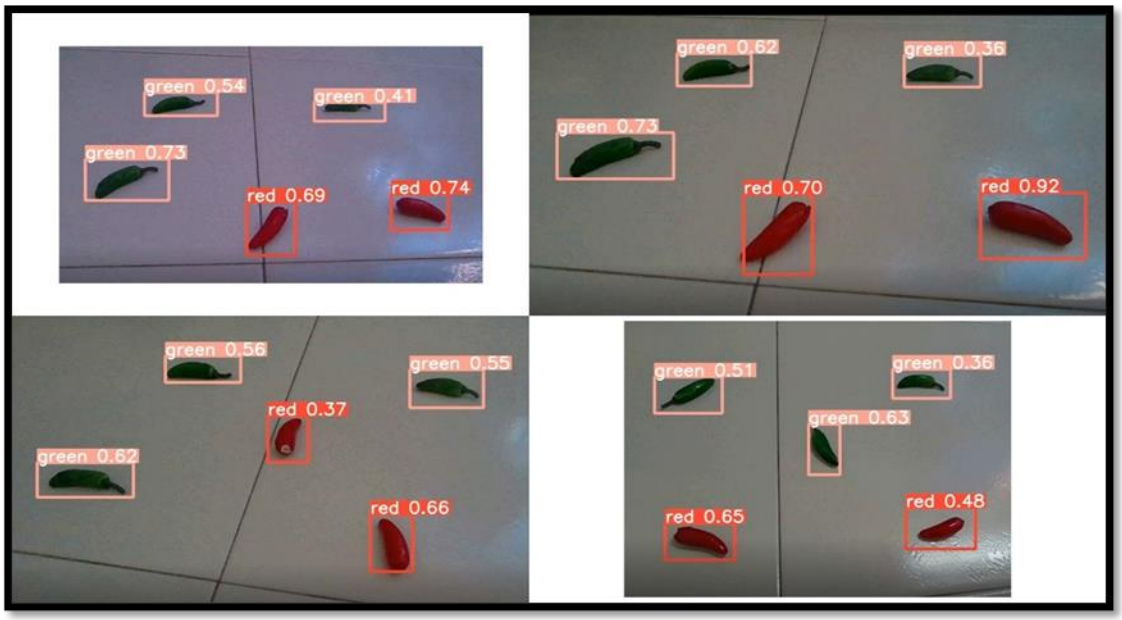

Figure 4.8: Result combined red and green chili

4.5 Result analysis on labeling chilis on chili plant

This result is obtained from labeling chili on chili plant. In this part, green and red chili is labeled at chili plant. By using the same pictures for these two types of chilies and also using batch 16, epoch 300 and also the weight is yolov5s for training the pictures, the results can be shown in Figure 4.9 and 4.10.

Figure 4.9: The label for red and green chili

Figure 4.10: Prediction for red and green chili

Figure 4.11: Train and Validation Images

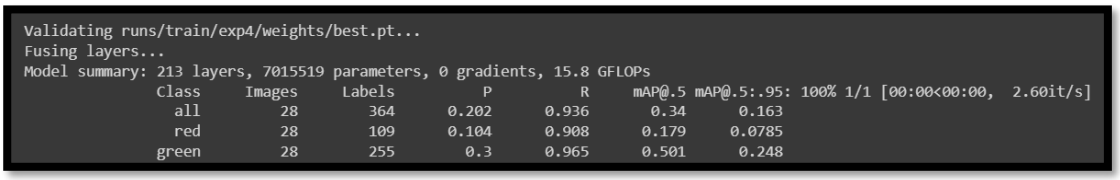

Figure 4.12: Accuracy for average red and green chili

As seen in the result prediction, for the accuracy of red chilies it is higher compared to green chilies. The same reason as mentioned in previous experiments in section 4.4 because the color of red chilliest is highly brighter than green chilies. However, these two chilies can be detected well and there is an interference due to the two chilies being nearby and there are unwanted objects such as trees and leaves that interfere with the labeling process. The total accuracy for red chilies is 0.179/17% while for green chilies it is 0.501/50%. It can be concluded that low accuracy is obtained when involving chili plant but the model can be localized well the fruits accordingly.

Figure 4.13: Result on chili plant for both red and green colors

4.6 Results analysis combination for 2 method labeling chili

This result is obtained from labeling chili on chili tree individually. Green and red chili is labeled at chili tree. By using the previous pictures for these two types of chilies and also using batch 16, epoch 100 and the weight is yolov5s for training the pictures, the results can be shown in Figure 4.14 and 4.15.

Figure 4.14: The label combination method for both red and green chilis

Figure 4.15: Prediction for combination method both red and green chilis

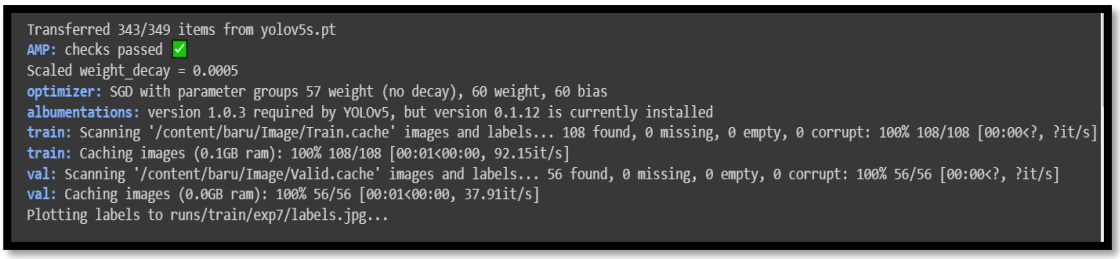

Figure 4.16: Train and Validation Images

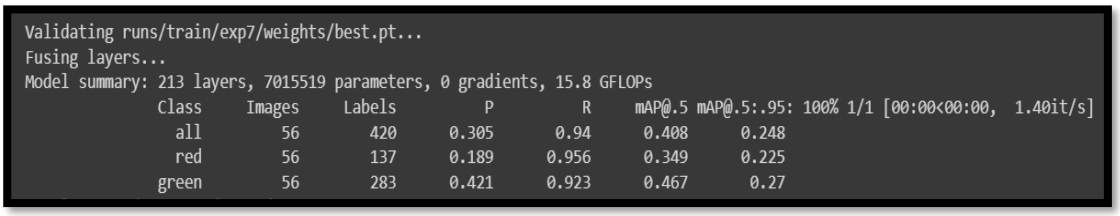

Figure 4.17: Accuracy for combination method both red and green chilis

As seen in the result prediction, for the accuracy of combination method red chilies is higher compared to green chilies. However, the combination method for the two chilies can be detected well and there is an interference due to the two chilies being nearby and there are unwanted objects such as trees and leaves that interfere with the labeling process. The total accuracy for combination method on red chilies is 0.349/34% while for green chilies is 0.467/46%.

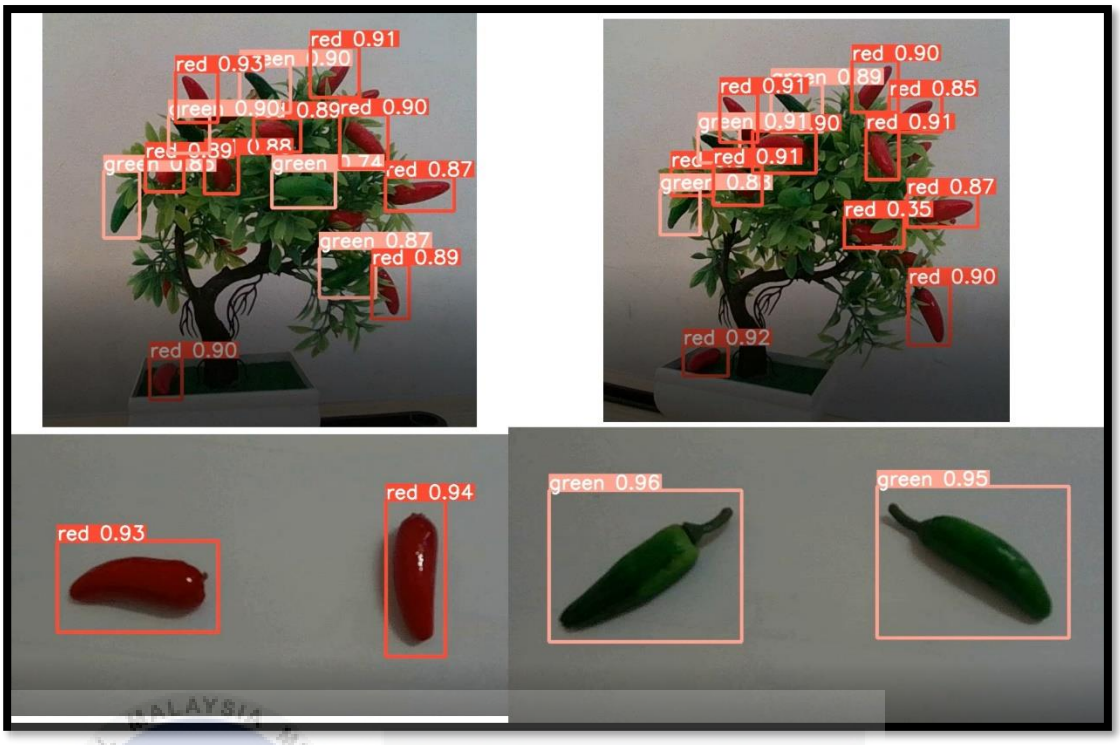

Figure 4.18: Result on combination for both method

CHAPTER 5

CONCLUSION AND FUTURE WORKS

In this chapter, overall achievement outcomes are being discussed and summarized. This includes some future recommendations for the continuation of this work.

5.1 Conclusion and future works

This work focuses on developing the most effective way for chili detection and localization by using YOLOv5. Using the makesense.ai in web browser to perform process of labeling chili fruits. Camera Intel Realsense has been used to collect the dataset and well-known object detection method YOLOv5 is applied to train and validate the labeled chili images for measuring the accuracy for detestation and localization at chili fruits on plant. For improvement, this work could be improve by expanding a dataset that contains an equivalent number of images for each class to avoid the occurrence of overfitting throughout the training process. Secondly, this

image detection can be implemented for real -time environment situation. This project unable to achieve the desired accuracy due to the reasons mentioned above. Lastly, the image for the data set needs to be properly cleaned and filtered so that the desired object (chili fruits in this case) can be detected more accurate without any interference.

REFERENCES

[1] "Intel® RealSense™ Depth Camera D400-Series (Intel® RealSense™ Depth Camera D415, Intel® RealSense™ Depth Camera D435)," 2017. [Online]. Available: www.intel.com/design/literature.htm.

[2] A. Kadambi, A. Bhandari, and R. Raskar, "3D depth cameras in vision: Benefits and limitations of the hardware with an emphasis on the first-and second-generation kinect models," in *Advances in Computer Vision and Pattern Recognition*, vol. 67, Springer London, 2014, pp. 3–26. doi: 10.1007/978-3-319-08651-4_1.

[3] H. M. Merklinger, "View Camera Focus and Depth of Field-Part II." [Online]. Available: www.mr-alvandi.com

[4] H. G. Jeon, J. Y. Lee, S. Im, H. Ha, and I. S. Kweon, "Stereo Matching with Color and Monochrome Cameras in Low-Light Conditions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Dec. 2016, vol. 2016-December, pp. 4086–4094. doi: 10.1109/CVPR.2016.443.

[5] S. Kumar, A. Balyan, and M. Chawla, "Object Detection and Recognition in Images," 2017. [Online]. Available: www.ijedr.org

[6] J. M. Buhmann, J. Malik, and P. Perona, "Image recognition: Visual grouping, recognition, and learning." [Online]. Available: www.pnas.org

[7] K. Murphy, A. Torralba, D. Eaton, and W. Freeman, “Object detection and localization using local and global features.” [Online]. Available: http://web.mit.edu/torralba/www/database.html

[8] A. Kamilaris and F. X. Prenafeta-Boldú, “A review of the use of convolutional neural networks in agriculture,” *Journal of Agricultural Science*, vol. 156, no. 3. Cambridge University Press, pp. 312–322, Apr. 01, 2018. doi: 10.1017/S0021859618000436.

[9] N. K. Delina, I. H. Ayub Wahab, and A. Khairan, “The Measurement of Maturity Level in Chili’s Using Index Pixel,” in *Journal of Physics: Conference Series*, Jul. 2020, vol. 1569, no. 2. doi: 10.1088/1742-6596/1569/2/022032.

[10] “Stereo and 3D Vision.”

[11] C. Qi, “Introduction to Deep Learning CS468 Spring 2017.”

[12] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, “Dive into Deep Learning Release 0.14.3,” 2020.

[13] Y. Tang *et al.*, “Recognition and Localization Methods for Vision-Based Fruit Picking Robots: A Review,” *Frontiers in Plant Science*, vol. 11. Frontiers Media S.A., May 19, 2020. doi: 10.3389/fpls.2020.00510.

[14] Grace Karimi, “Introduction to YOLO Algorithm for Object Detection,” *Section*, Apr. 15, 2021.

[15] S. D. R. B. G. A. F. Joseph Redmon, “You Only Look Once: Unified, Real-Time Object Detection,” *SEMANTIC SCHOLAR*, Jun. 08, 2015.

[16] L. Zhong and Q. Zou, “YOLO: You Only Look Once Unified Real-Time Object Detection.”

[17] A. F. Joseph Redmon, “YOLO9000: Better, Faster, Stronger,” *Cornell University*, Dec. 25, 2016.

[18] S. Gupta, T. Uma, [D., and] Student, “YOLOv2 based Real Time Object Detection,” *International Journal of Computer Science Trends and Technology*, vol. 8, 2013, [Online]. Available: www.ijcstjournal.org

[19] Venkata Krishna Jonnalagadda, “Object Detection YOLO v1 , v2, v3,” *Mediuam*, Jan. 31, 2019.

[20] Vidushi Meel, “YOLOv3: Real-Time Object Detection Algorithm,” *viso.ai*.

[21] “Introduction To YOLOv4,” *Neelam Tyagi*, Jun. 25, 2020.

[22] Roman Orac, “What’s new in YOLOv4,” *Medium*, May 19, 2020.

[23] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.10934

[24] Mathias Gudiksen, “Getting started with YOLOv5,” *Medium*, Sep. 12, 2021.

[25] “How to Use Yolo v5 Object Detection Algorithm for Custom Object Detection.” [Online]. Available: https://github.com/ultralytics/yolov5

[26] P. Lihi Gur Arie, “The practical guide for Object Detection with YOLOv5 algorithm,” *Medium*.

[27] Jacob Solawetz, “YOLOv5 New Version - Improvements And Evaluation,” *roboflow*, Jun. 29, 2020.

[28] “Neural Nets: a particularly useful Black Box.”

[29] “ANN vs CNN vs RNN | Types of Neural Networks.” https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/ (accessed Jan. 08, 2022).

[30] “Intel® RealSense™ Technology Powers Machines That Can See.”

[31] W. C. Seng and S. H. Mirisaee, “A New Method for Fruits Recognition System.”

[32] K. W. Hsieh *et al.*, “Fruit maturity and location identification of beef tomato using R-CNN and binocular imaging technology,” *Journal of Food Measurement and Characterization*, vol. 15, no. 6, pp. 5170–5180, Dec. 2021, doi: 10.1007/s11694-021-01074-7.

[33] A. C. Sari, H. Setiawan, T. W. Adiputra, and J. Widyananda, “FRUIT CLASSIFICATION QUALITY USING CONVOLUTIONAL NEURAL NETWORK AND AUGMENTED REALITY,” *J Theor Appl Inf Technol*, vol. 99, p. 22, 2021, [Online]. Available: www.jatit.org